APPENDIX H

INFORMIX-STRUCTURED QUERY LANGUAGE (I-SQL) (TEST DOMAIN)

TABLE OF CONTENTS

Blank Page

APPENDIX H

INFORMIX-Structured Query Language (I-SQL) (Test Domain)

H.1 General. In addition to the standard access to either the OSCSC (Standard Army Retail Supply System-Gateway [SARSS-GW] in continental United States [CONUS]), OSCEUR (SARSS-GW Europe), or OSCPAC (SARSS-GW Pacific) production domains, installation system managers are given shell access to the test domain. This access has been granted to give system managers the ability to run structured queries against selected database tables for their installations.

NOTE: Objective Supply Capability (OSC) has undergone a name change and is now called the SARSS-GW. All references to OSC and gateway have been changed or refer to SARSS-GW.

H.2 Requesting Log-In IDs and Passwords. Use the procedures outlined in appendix C for requesting log-in IDs and passwords.

H.3 Accessing the Test Domain. System managers should follow the instructions in this appendix for accessing the test domain. Establishing communications is the same as when connecting to either OSCSC, OSCEUR, or OSCPAC, with the exception of the net address. Use 144.251.20.4 for the test domain.

H.4 Available Tables. The following tables are available:

abf_tab	group_info_tab	rfo_rpt
abf_xref_tab	hier_tab	trans_hist_tab
dodaac_tab	instin_tab	trans_out_tab
doc_hist_tab	msc_tab	unit_out

NOTE: The database administrator (DbA) at the Defense Mega Center (DMC) updates these tables daily.

H.5 Introduction. I-SQL is the database management system SARSS-GW uses. The following information is a guide for using I-SQL.

a. Overview. I-SQL is a database management system based on the relational model of database design and on the SQL, which is specifically written for database applications.

(1) SQL was developed by IBM as a prototype to model relational algebra, the underlying mathematical model for relational databases. It has since been used in numerous software packages, and has become an industry standard database language.

(2) I-SQL is based on the relational database SQL. I-SQL is written in C programming language, and currently runs under Virtual Memory Storage (VMS), UNIX, Xenix, Microsoft-Disk Operating System (MS-DOS), local area networks, and other proprietary operating systems.

b. Database structure. A database is an organized collection of information. The mechanism that controls how information is added, changed, deleted, and retrieved from data files is called a database management system (DBMS). Data is stored in files. Each file contains information about a particular category of information (for example, stock numbers and customers). Within each table, data can be divided into columns and rows.

(1) A column may represent an individual topic of information, such as a last name or a unit name in a table.

(2) A row represents the information for a single entity across all columns. In a customer table, a row represents all the information for a single customer.

(3) Figure H.5-1 represents database organization.

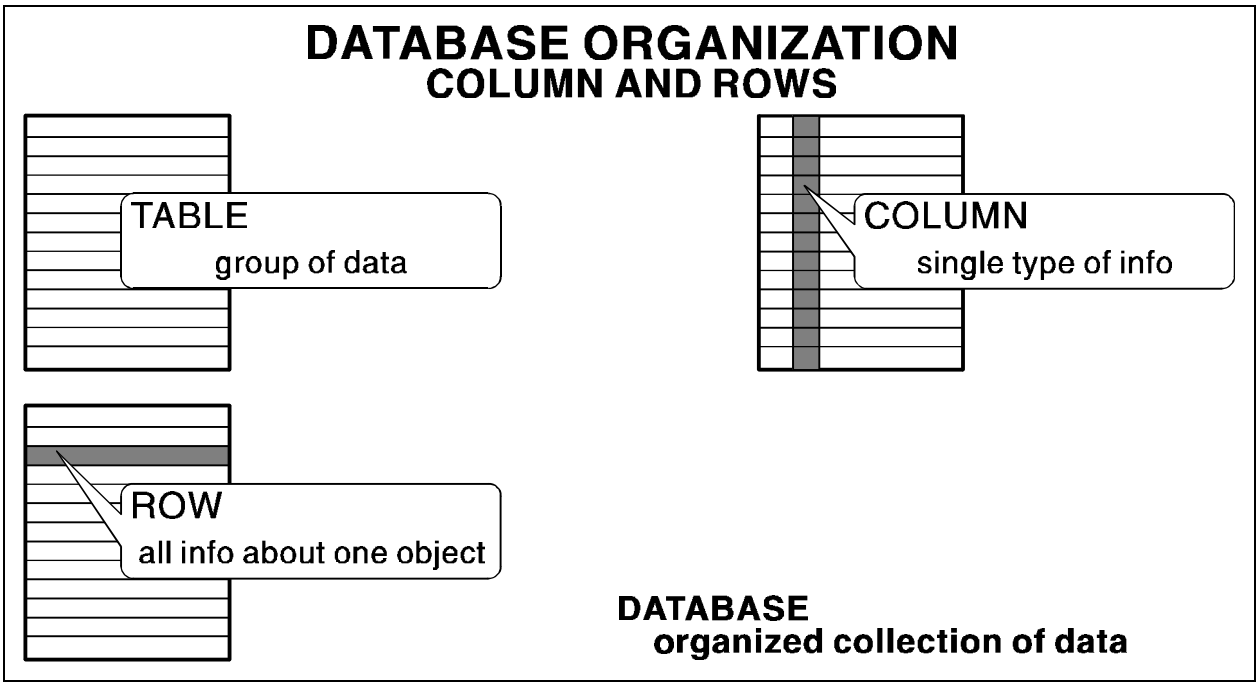

Figure H.5-1. Database Organization

c. Relational database. SQL is a true relational database management system that contains all the essential sustainment programs and documentation needed to develop, install, and preserve custom database applications. In a relational model, the tables are not linked in any way. Instead, the tables are logically joined through common columns of information. The organization of a database into tables, columns, and rows depends mainly on the nature of the data itself, rather than how the data is used.

(1) There are two types of column information:

(a) Information of interest about that category (for example, stock number or customer).

(b) Key information that identifies each row in that table.

(2) There are two types of relationships between tables:

(a) One-to-many. For each row in the one table, there may be any number of associated rows in the many table, but for each row in the many table, there is only one associated row in the one table.

(b) Many-to-many. Each row in either table may be associated with any number of rows in the other table. Many-to-many relationships are generally handled by creating a third table and using it to generate two, one-to-many relationships that logically join the first two tables through the third table.

d. Using a database. Using a database requires you to perform certain actions.

(1) The first task is to create the database and associated tables. I-SQL provides a menu system to simplify this task. The database can be created by providing a name and executing a single menu option.

(2) One of the functions of a DBMS is to provide a means by which users have easy, yet controlled access to a database. This is commonly accomplished through the use of data entry screen forms. I-SQL has a screen form generator called Perform. The Perform utility lets the user create a variety of data screens with various controls over data input and screen display.

e. Query language interpreter. INFORMIX-SQL includes an option that lets users interact with the database without predefined forms or reports.

f. User-menu system. Once screen forms and reports are created, users need an easy method to access them. One method is to create a menu system to call up any form or report when the user presses a key. This method is called the user-menu

system.

H.6 Selecting a Database. Follow these steps when selecting a database.

a. Starting I-SQL. To start the I-SQL program, type isql at the operating system prompt $.

(1) The INFORMIX-SQL Main Menu, (figure H.6-1) appears

```
INFORMIX-SQL:  Form Report Query-Language User-menu Database Table  Exit
Run, Modify, Create or Drop a form.

-----------------------------------Press CTRL-W for Help----
```

Figure H.6-1. I-SQL Main Menu

(2) Table H.6-1 explains the I-SQL Main Menu options.

Table H.6-1
I-SQL Main Menu Options and Explanation

Options	Explanation
Form	This option lets you create and run Perform screen forms.
Report	This option lets you create and run customized ACE reports.
Query-Language	This option lets you use the I-SQL Query Language interpreter to write and execute statements.
User-menu	This option lets you create and run a customized menu system to access forms, reports, system commands, etc.
Database	This option lets you create, select, or drop a database.
Table	This option lets you create, alter, drop, or get information about a table.
Exit	This option returns you to the operating system.

b. I-SQL Menu format. I-SQL menus use this format:

(1) The menu name is at the top left, followed by a colon.

(2) The options available for that menu are listed to the right of the menu name.

(3) Comments associated with each menu item are listed on the second line.

c. Selecting menu options. You can select a menu option either by typing the first letter of the option name or by using the arrow keys or space bar to move the cursor to the desired option and pressing <Enter>.

(1) After you select the initial option, some options display a >> on the top line to indicate that more information, such as the name of a form or report, must be entered. Enter text either by typing in the appropriate name and pressing <Enter> or by using the arrow keys or space bar to move the cursor to the desired name in the list and then pressing <Enter>. (To cancel the command, press <Del> or <Control><C>.)

(2) If a menu contains more options than can fit on one screen, three dots appear at the end of the line. You can view the additional options by pressing the arrow keys or space bar.

(3) The last option on the menu is always Exit. Exit always takes you back one level from where you are. In the case of the Main Menu, Exit will return you to the operating system.

(4) Press <Control><W> for on-line help.

d. Selecting a database. Use the Database option on the I-SQL Main Menu (figure H.6-1) to select databases. Type D to select Database.

(1) The I-SQL Database Menu (figure H.6-2) appears

```
DATABASE: Select  Create  Drop  Exit
Select a database to work with.

-----------------------------------Press CTRL-W for Help----
```

Figure H.6-2. I-SQL Database Menu

(2) Table H.6-2 explains the I-SQL Database Menu options.

Table H.6-2
I-SQL Database Menu Options and Explanation

Options	Explanation
Select	This option lets you select an existing database as the current database.
Create	This option lets you create a new database.
Drop	This option lets you drop the existing database and remove all its files.
Exit	This option returns you to the Main Menu.

e. Selecting the SARSS-GW database. Type S for Select.

(1) The databases available appear on the screen. Only three databases should be available: OSCSC, OSCEUR, and OSCPAC.

(2) Press the arrow key or the space bar to move to your choice (if more than one choice is available).

(3) Press <Enter>.

(4) A screen similar to that in figure H.6-3 appears.

```
INFORMIX-SQL:  Form Report Query-Language User-menu Database Table  Exit
Run, Modify, Create, or Drop a form.

----------------oscsc----------------Press CTRL-W for Help----
```

Figure H.6-3. I-SQL Main Menu

(5) Type E for Exit to return to the Main Menu.

H.7 The Perform Screen Generator. Perform is an I-SQL program that lets you enter data into the database using an interactive screen form. The screen form is generally the easiest way to enter information. Much like a printed form, the screen form contains blank spaces you can fill in. The blank spaces, called fields, usually correspond to columns in your tables

a. Creating a form. You can create, compile, and run a form through the I-SQL menu system.

(1) On the Main Menu, type F for Form.

(2) The screen in figure H.7-1 appears.

```
FORM: Run  Modify  Generate  New  Compile  Drop  Exit
Use a form to enter data or query a database.

----------oscsc----------Press CTRL-W for Help----
```

Figure H.7-1. I-SQL Form Menu

(3) Table H.7-1 explains the I-SQL Form Menu options.

Table H.7-1
I-SQL Form Menu Options and Explanation

Options	Explanation
Run	This option executes an existing compiled form.
Modify	This option lets you edit an existing form.
Generate	This option creates and compiles a default form based on one or more tables.
New	This option lets you create a new form specification from scratch.
Compile	This option compiles a form specification that you have created or modified.
Drop	This option deletes an existing form specification.
Exit	This option returns you to the I-SQL Main Menu.

b. Creating and compiling a default form. You can have the system create a default form. The menu will display prompts, and you will make entries accordingly. Figure H.7-2 shows you some sample menu prompts you may encounter in the process and some corresponding entries you can make.

Menu Prompts	Your Response or Entries
SELECT Database >>	oscsc
GENERATE FORM >>	hier (example)
CHOOSE TABLE >>	hier_tab (example)

Figure H.7-2. Sample Default Form Menu Prompts and Entries

(1) On the Form Menu, type G for Generate.

(a) Generate creates both source and executable files for the tables you specify.

(b) The default generator lets you create default forms that access two or more tables from the database. This example uses only one database table.

(2) A screen like the one in figure H.7-3 appears.

```
GENERATE FORM:  Table-selection-complete  Select-more-tables  Exit
Continue creating a default form with the selected tables.
```

Figure H.7-3. I-SQL Generate Form Menu

(3) Enter a name for the form you wish to create and press <Enter>.

(4) Press the arrow key to move the cursor to the desired choice (if there is more than one choice available) and press <Enter>.

(5) Type T for Table-selection-complete.

(6) A ready-to-use default form is created, and the screen in figure H.7-4 appears.

```
PERFORM: Query  Next  Previous  View  Add  Update  Remove  Table  ...
Searches the active Database table.          ** 1: hier_tab table**
hier_indic          [              ]
abf_id              [              ]
abf_hier            [              ]
hier_order          [      ]
```

Figure H.7-4. I-SQL Perform Screen Displaying the Default Form

(7) Figure H.7-4 shows a sample default form created by the default form generator.

c. Running a form. You can run a form once it has been created and successfully compiled.

(1) To run the form, type R for Run on the I-SQL Form Menu (figure H.7-1).

(2) The Run Form screen appears.

(3) Use the arrow keys to choose a form, or enter a form name, then press <Enter>.

(4) The Perform screen appears.

(a) All of the options will not fit onto one screen; therefore, two screens are available.

(b) The screens are shown in figures H.7-5 and H.7-6.

```
PERFORM: Query Next Previous Add Update Remove Table Screen ...
Searches the active Database table.
```

Figure H.7-5. I-SQL Perform Menu (Screen 1)

```
PERFORM: Current Master Detail Output Exit
Displays the current row of the current table.
```

Figure H.7-6. I-SQL Perform Menu (Screen 2)

(c) To get from one screen to the next, use the arrow keys.

(5) Table H.7-2 explains the I-SQL Perform Menu options.

Table H.7-2
I-SQL Perform Menu Options and Explanation

Options	Explanation
Query	This option retrieves selected rows from the current table and displays the first row. The number of rows retrieved is displayed at the bottom of the screen.
Next	This option displays the next row retrieved. You can move several rows ahead by preceding *Next* with a number (3n moves three rows ahead).
Previous	This option displays the row previously retrieved. You can move several rows back by preceding Previous with a number (3n moves three rows back).
Add	This option lets you add a row.
Update	This option lets you update the row if changes were made and you have permission.
Remove	This option lets you delete a row if you have permission. Before the row is deleted, a verify prompt appears.
Table	This option changes the current table (only applies if more than one table is specified in the form).
Screen	This option displays the next screen (only applies if there are multiple screens).
Current	This option displays the current row with update information from the table.
Master	This option makes the Master Table the current table and queries based on the join information (applies only if a Master/Detail relationship is specified).
Detail	This option makes the Detail Table the current table and queries based on the join information (applies only if a Master/Detail relationship is specified).
Output	This option lets you send the output from a query to a system file.
Exit	This option returns you to the I-SQL Form Menu.

d. Querying the database. You can query the database by typing the search criteria into the appropriate fields and executing a search.

(1) Notice that the query option is highlighted.

(2) Using the form you just created, press <Enter> to query the database.

(3) The cursor drops to the first field on the query screen.

(4) To query specific data on the table, type a specific item in the desired field and press <Esc> to execute.

(5) Figure H.7-7 shows the results of a query using 62 in the abf_id field.

```
PERFORM: Query Next Previous View Add Update Remove Table Screen ...
Shows the next row in the Current List.          ** 1: hier_tab table**
hier_indic              [1              ]
abf_id                  [62             ]
abf_hier                [102             ]
hier_order              [1      ]

1 row(s) found
```

Figure H.7-7. I-SQL Perform Screen Displaying Results of a Query

H.8 Query Language. SQL is the basis on which I-SQL is formulated.

a. Operators and wild cards. SQL lets you use special relational operators and wild cards in SQL statements. Table H.8-1 contains examples of some operators and wild cards.

Table H.8-1
Query-Language Operators and Wild Cards and Explanation

Operator/Wild Cards	Action
>	Finds values greater than.
<	Finds values less than.
<>	Finds values not equal to.
>=	Finds values greater than or equal to.
<=	Finds values less than or equal to.
X:Y	Finds values X through Y.
XY	Finds value X or Y.
<<	Finds the first row.
>>	Finds the last row.
Z*	Finds values that start with Z.
[Zz]*	Finds values that start with Z or z.
Z??	Finds three-character values that start with Z.
=*	Finds the value of *.
=	Finds a null value.
BLANK	Finds all values from the table.

b. Using the interpreter. Statements can be written and executed using the I-SQL Query-Language editor and interpreter.

(1) To access the SQL Menu, type Q for Query-Language on the I-SQL Main Menu.

(2) The screen in figure H.8-1 appears.

```
SQL: New Run Modify Use-editor Output Chose Save Info Drop Exit
Enter new SQL statements using the SQL editor.

----------oscsc----------Press CTRL-W for Help----
```

Figure H.8-1. SQL Menu

(3) Table H.8-2 explains the SQL Menu options.

Table H.8-2
SQL Menu Options and Explanation

Options	Explanation
New	This option accesses the editor so one or more statements may be entered. Use the <Esc> key to exit.
Run	This option executes the current statements.
Modify	This option lets you change the current statement.
Use-editor	This option lets you select a system editor.
Output	This option sends the statement output to a printer, file, or pipe.
Save	This option saves the current statements in a command file, which you name. The extension .sql is automatically appended.
Choose	This option presents a list of files that end with the extension .sql. When chosen, the contents of this file become the current statement.
Info	This option displays information about a table, also available from the Table Menu.
Drop	This option lets you drop or delete a command file.
Exit	This option returns you to the I-SQL Main Menu.

c. Formatting the SQL statement. You may format an SQL statement almost any way you want.

(1) You can put every word on a different line, or as much as you can fit on one line. For example, this query:

```
select
    dodaac,
    abf_id,
    msc

from abf_xref_tab
```

is the same as this query:

```
select dodaac, abf_id, msc from abf_xref_tab
```

(2) You may include more than one SQL statement at one time; however, you must separate each statement using a semicolon (;).

d. Searching for rows and columns: the select statement. The Select statement is the basis for all queries. Use it to query the database and display the results on screen.

(1) You can use seven clauses in the Select statement. The following four will be discussed:

```
Select      (required)
From        (required)
Where
Order By
```

(2) The Select and From clauses are required; the other clauses are optional.

(3) You may view one, several, or all rows in a table. Unless otherwise specified, the Select statement finds and displays all rows.

(4) The format for finding all columns and rows in a table is:

```
select *
from msc_tab
```

which is the same as:

```
select * from msc_tab
```

(5) After you enter the Select statement, press <Esc>.

(6) Choose the Run option from the menu. This is the simplest form of SQL statement. It displays all rows and columns found in the msc_tab database table.

(7) SQL displays as many rows as will fit on the screen. If all rows do not fit on one screen, you can select the Next option to scroll through any additional screens you may need.

(8) When the last row appears on the screen, the number of rows retrieved appears at the bottom of the screen. Output from this query is similar to that in figure H.8-2.

```
PERFORM: Query  Next  Previous  View  Add  Update  Remove  Table
Screen ...
Searches the active database table.          ** 1:  msc_tab table**
msc             [1            ]
msc_name        [1st CAV DIV]
fc_inter_1      [   ]
fc_inter_2      [   ]
fc_inter_3      [   ]
fc_inter_4      [   ]
fc_inter_5      [   ]
fc_inter_6      [   ]
fc_inter_7      [   ]
fc_inter_8      [   ]
fc_inter_9      [   ]
fc_inter_10     [   ]
fc_ext_1        [FG]
fc_ext_2        [   ]
fc_ext_3        [   ]
fc_ext_4        [   ]
fc_ext_5        [   ]
fc_ext_6        [   ]
fc_ext_7        [   ]
fc_ext_8        [   ]


123 row(s) retrieved
```

Figure H.8-2. Select From Example

(9) You may also search for specific columns within a table. Instead of using the * in the Select line, you may define only those columns you wish to view:

```
select msc, msc_name
from msc_tab
```

(10) This statement shows only the columns msc and msc_name for all rows in the msc_tab.

(11) Use commas to separate all columns you wish to select.

(12) To execute any query, press <Esc>. Then, select the Run option from the menu. See the sample output in figure H.8-3.

```
RUN:  Next  Restart  Exit
Display the next page of query results.

------------------------------oscsc----------------Press CTRL-W for Help-------------------------

     msc      msc_name

     287      24TH QMS WEG
     288      25TH FSB WEL
     289      85TH MAINT WFW
     290      DOL RTM WEH
     130      INSTLN SAILS HU
     131      11TH SIGNAL HU
     150      SAILS KX
     151      19TH ENGR BT DS
     152      COMBAT HET PLT
     153      101ST PSC
     180      DOL FT LEE
     181      DUMMY DSU
     182      267TH QM CO
     183      555TH MP CO



     14 row(s) retrieved
```

Figure H.8-3. Select (Column) From Example

(13) You may also perform calculations using the aggregate functions listed in table H.8-3:

Table H.8-3
Aggregate Functions

Count (*)	Counts the number of rows retrieved.
Sum	Adds the values of specified numeric columns.
Avg	Averages the values of specified numeric columns.
Max	Finds the maximum value of a column you specify.
Min	Finds the minimum value of a column you specify.

(a) When using the Sum, Avg, Max, and Min functions, enclose the column name in parentheses.

(b) If you include more than one of these functions in the Select clause, separate the functions using commas.

(c) Do not specify a column name with the Count(*) function.

(14) For example, you could find the average, minimum, and maximum amount for the column expended in the fund_tab with this Select statement:

```
select avg(expended), min(expended), max(expended)
from fund_tab
```

(15) The result is similar to that shown in figure H.8-4.

```
SQL: New Run Modify Use-editor Output Chose Save Info Drop Exit
Enter new SQL statements using the SQL editor.

----------oscsc----------Press CTRL-W for Help----

  (avg)          (min)          (max)

82823.40      14227.78    123998.19


1 row(s) retrieved
```

Figure H.8-4. Select (Calculated) From Example

e. Searching for specific rows. To search for specific in a table, include a Where clause in the Select statement.

(1) Use the Where clause to specify the search criteria used to determine which rows to retrieve.

(2) The format for a query using the Where clause is:

```
select *                    {This selects all columns}
from fund_tab          {from the fund_tab}
where allocated = 300000   {limit search}
```

(3) The result is similar to that shown in figure H.8-5.

```
SQL:  New  Run  Modify  Use-editor  Output  Chose  Save Info  Drop Exit
Enter new SQL statements using the SQL editor.

----------------------oscsc----------------Press CTRL-W for Help-----------

   msc      fund_cd     allocated      credited     expended     Typ

    61         yy       300000.00      1500.00     123998.19      1

    61         yy       300000.00        10.00      46598.30      2
2 row(s) retrieved
```

Figure H.8-5. Select From Where Example

(4) Rules for using the relational operators in a Select statement with a Where clause are:

(a) Follow the Where keyword with search conditions or descriptions of the rows you want to find.

(b) Enclose Char and Date values in quotation marks when the search condition includes a column name, a relational operator, and a value.

(5) You may use more than one search condition, but the conditions must be separated by the and or the or clauses. For example, the statement:

```
select *
from fund_tab
where credited = 0 and expended > 100000
```

(6) The result is similar to that shown in figure H.8-6.

SQL: New Run Modify Use-editor Output Chose Save Info Drop Exit
Enter new SQL statements using the SQL editor.

-----------------------oscsc----------------Press CTRL-W for Help-----------

msc	fund_cd	allocated	credited	expended	Typ
61	xx	140000.00	0.00	110784.23	1
61	xx	140000.00	0.00	125723.40	2

2 row(s) retrieved

Figure H.8-6. Select From Where Example

(7) Use the Matches keyword to find rows containing a particular word or phrase.

(a) Like the = relational operator, the Matches clause is better suited to search for nonnumeric data.

(b) The Matches clause may only search for character type data.

(8) When using Matches, enclose the search criteria in quotation marks. The strength of the Matches keyword lies in three wild card characters (table H.8-4).

Table H.8-4
Matches Wild Card Characters

Symbol	Meaning
*	Matches zero or more characters.
?	Matches any single character.
[...]	Matches any one of a specified string of characters. The string can be implied by way of a range.

(9) Examples of the Matches keyword using wild card characters are shown in table H.8-5.

Table H.8-5
Matches Using Wild Cards

Matches "b*"	Finds any character column that begins with a lowercase b.
Matches "*CE"	Finds any character column that ends in an uppercase CE.
Matches "*ace*"	Finds any character column that contains the letters ace.
Matches "mak?"	Finds any character column containing four characters that begin with the letters mak.
Matches "[a-c]*"	Finds any character column beginning with an a through c.

(10) The statement below shows an example of Matches using wild cards:

```
select fund_cd, allocated, expended
from fund_tab
where fund_cd matches "*AD"
```

(11) The result is similar to that shown in figure H.8-6.

```
SQL:  New  Run  Modify  Use-editor  Output  Chose  Save Info  Drop Exit
Enter new SQL statements using the SQL editor.

-----------------------oscsc----------------Press CTRL-W for Help-----------

                  fund_cd        allocated                        expended

                    AD          170000.00                         14227.78

                    AD          140000.00                         10784.23
2 row(s) retrieved
```

Figure H.8-6. Select From Where Matches Example

(12) You may use the Not option with matches as well (Not Matches), which makes the search condition true when the selected character column does not match the specified search pattern.

f. Sorting query output. Use the Order By clause to sort query results by any column chosen in the Select clause.

(1) The statement below shows the Order By clause with the expended column:

```
select *
from fund_tab
where typ = 1
order by expended
```

(2) The result is similar to that shown in figure H.8-7.

```
SQL: New Run Modify Use-editor Output Chose Save Info Drop Exit
Enter new SQL statements using the SQL editor.

----------oscsc----------Press CTRL-W for Help----------

msc          140
fund_cd      AD
allocated    $20000000.00
credited     $0.00
expended     $0.00
typ          1

msc          505
fund_cd      AC
allocated    $104464.35
credited     $0.00
expended     $7184.83
typ          1

2 Row(s) Retrieved
```

Figure H.8-7. Select From Order by Example

(3) You may use more than one column for sorting. Sorting is done in the sequence the columns are identified after the Order By clause.

g. Outputting query results. You may send the results of your query to a file for reviewing or printing.

(1) Write the query and select the Output option on the menu. The Output Menu appears (figure H.8-8).

```
OUTPUT: Printer New-file Append-file To-pipe Exit
Send query results to New-file.

----------oscsc----------Press CTRL-W for Help----
```

Figure H.8-8. Output Menu

(2) To append to an existing file, select the New-file option to create a new file, or the Append-file option.

(3) You may print the file after it is created.

(4) You must have write permissions in the directory to which you send the file (see appendix I for permission and printing information).

H.9 The ACE Report Writer. You can use the I-SQL ACE Report Writer to customize output from a default format. ACE is an I-SQL program that lets you retrieve and format information from a database in a style you design. Use it to generate management reports, mailing lists, form letters, etc.

a. Report Menu. You can create, compile, and run a report specification from the I-SQL Report Menu.

(1) On the I-SQL Main Menu, type R for Report.

(2) The screen in figure H.9-1 appears.

```
REPORT: Run Modify Generate New Compile Drop Exit
Run a report.

----------oscsc----------Press CTRL-W for Help----
```

Figure H.9-1. I-SQL Report Menu

(3) Table H.9-1 explains the I-SQL Report Menu options.

Table H.9-1
I-SQL Report Menu Options and Explanation

Options	Explanation
Run	Runs a report.
Modify	Lets you edit an existing report.
Generate	Creates and compiles a default report.
New	Lets you create a new report specification from scratch.
Compile	Compiles a report specification that you have created or modified.
Drop	Deletes an existing report specification.

Exit	Returns you to the I-SQL Main Menu.

b. Generating a default report. A default report lists all fields in a table or tables you select in the order they appear on the table.

(1) To generate a default report, type G for Generate on the Report Menu.

(2) I-SQL prompts you for information.

(3) Table H.9-2 shows sample menu prompts and responses you can make.

Table H.9-2

Menu Prompts	Responses (Entries) You Can Make
SELECT Data Base >>	oscsc
GENERATE REPORT >>	(reportname)
CHOOSE TABLE >>	msc_tab (example)

d. Report specifications. All reports contain specifications, three of which are required and the other three are optional.

(1) The required specifications are Database, Select, and Format.

(a) Database identifies the database you want to access for information in the report.

(b) Select identifies the data you want in the report. You can enter multiple Select statements.

(c) Format specifies how you want to display the output.

(2) The optional specifications are Define, Input, and Output.

(a) Define defines local variables you want to use in the report.

(b) Input generates prompts for user-entered information in interactive reports.

(c) Output specifies the page margins and where the output is to be sent.

e. Procedural statements. ACE allows a variety of procedural, formatting, date, and aggregate statements to prepare reports. The list in table H.9-3 explains the statements that can be used in the Format specifications section of a report.

Table H.9-3
Statements for ACE Format Specifications

Statement	Explanation
Clipped	This removes trailing spaces from character data.
Column #	This moves the print head to a particular column on the page.
[Group] Count	This counts the number of rows returned by the Select statement. Group Count counts the groups of rows from an After Group of control block.
If...Then...Else	If evaluates an expression and takes conditional actions. If an action involves more than one statement, all the action statements must be between the key words Begin and End.
Let *Var-Name=Expr*	Let assigns a value to a defined variable.
Pageno	This is a key word that typically equals the current page number and is used in a header or trailer to number pages.
Pause [String]	The string is for an optional comment. Pressing return begins scrolling again. This command is ignored if the output does not go to a screen.
Print [Expr-List] [;]	This prints the contents of the expr-list. The expr-list can include both literal text and variable names separated by commas. PRINT has an implicit line feed that can be suppressed by ending the statement with a semi-colon.
Print File *Filename*	This prints the contents of filename into the report.
Skip *Integer* Lines	This skips the indicated number of lines in the report.
Skip to Top of Page	This causes subsequent printing to begin at the top of the next page.
Num-Expr Spaces	This skips the indicated number of spaces in the current line.
Expr1 Using *Expr2*	This displays *expr1* in the format designated by *expr2*. *Expr2* refers to a set of characters with special formatting meaning (for instance, <$#@>). Using applies to numeric and data.
Today	This is today's date.
Year (Date-Expr)	Year returns an integer that represents the year of the date-expression.

Table H.9-3 Statements for ACE Format Specifications	
Month (Date-Expr)	Month returns an integer corresponding to the month (1-12) of the date-expression.

Blank Page